DQN Algorithm

State representation → Deep NN

Convert state into numerical representation

Takes state as input and outputs action-values for each action

→ Experience Replay → Q-Learning Update

Store agent's experiences in a replay memory buffer

Updates NN weights using the loss function

→ Exploration / Exploitation → Target Network

Markov Decision Process

1. Agent: the system that is operated
2. Environment: the real-world environment with which the agent interacts
3. State: current state of the world (could be a finite or infinite set of states)
4. Action: what the agent does to interact with the environment (finite or inf. set)
5. Reward: positive or negative reinforcement the agent receives from the environment as a result of its actions.

- Return: total reward over all time-steps

  $Return = r_0 + \gamma r_1 + \gamma^2 r_2 + \ldots + \gamma^n r_n$ (γ = discount factor, r_n = reward)

- Policy: the strategy followed to pick an action
  - Deterministic Policy - policies where agents always choose the same fixed action when it reaches a particular state
  - Stochastic Policy - policies where the agent varies the actions it chooses for a state based on some probability for each action.